A Receptor-Centric Model for Emergence Phenomenon

A Receptor-Centric Model for Emergence Phenomenon

Michael Ji, mji@ed.ac.uk
School of Informatics, University of Edinburgh

Abstract

In the past few decades, there has been no breakthrough in the studies of emergence phenomenon. Most of the existing research work considers that an emergence appears when a system has a lot of self-organized parts; and believes that an emergent characteristic belongs to the system "as a whole", not to any individual part. Up to date, we still have no clear ideas about why and when an emergence would appear from a system. One of the main reasons is the lack of a theoretical computing model with which we can apply qualitative and quantitative reasoning. To address this issue, we propose a distributed Receptor-Centric Emergence (RCE) model, which consists of Receptor Agents, Actor Agents, Messages and an Interconnect. (1) A receptor receives messages from an environment, and applies an intrinsic control dynamics to update its internal states. (2) An actor periodically sends messages to the environment and exchanges data, imposing influences on other agents. (3) The interconnect is an asynchronous communication channel for all agents. (4) Any signal being secreted on the interconnect is encapsulated as a structured message. RCE's main idea is that, instead of belonging to the system as a whole, an emergent characteristic is actually an intrinsic state/feature of a receptor; an emergence appears only when a receptor senses/receives "enough" messages from the parts of the system/environment. In this paper, we first define the "Receptor-Centric Emergence" as "an emergence appears iff a receptor's state becomes expressed". Then, we propose both the Receptor State-Upgrading and the Emergence-Decider algorithms, with which we can answer the "why an emergence appears" question. Also, we novelly describe the semantics of "as a whole" from RCE's perspective; then we prove that, if the effect of the messages sent by each actor to a receptor is *influence*, and the threshold for the receptor to flip this state's status is *threshold*, an emergent characteristic would appear when the total number of actor agents reaches the $\log_{(1-influence)}(1-threshold)$.

Distributed Characteristics of Emergent Behaviors

We consider that a complex system is essentially a distributed computing environment, composed of many parts that connect with each other. In order to understand why and when an emergent behavior would occur, we define a REC model is composed of at least a receptor agent, multiple actor agents, an interconnect and the messages. Investigating emergent behaviors from a distributed computing perspective has many advantages. For example, we can apply formal symbolic approaches to design emergence-expression algorithms. Also, we can take full advantage of all the theoretical results from distributed computing research community. For instance, the RCE model satisfies all the principles of distributed systems, such as CAP, FLP and all other impossibility results.

Receptor-Centric Emergence Model

Receptor-Centric Emergence Model	
Components	Description
Receptor	An agent that can receive messages of different types from actor agents, and applies an intrinsic control dynamics to update its internal states.
Actor	An agent that periodically sends out one type of messages to the environment. An actor may be defective and terminated accidentally.
Interconnect	An asynchronous network that is able to forward messages sent by actor agents with best efforts. There is no guarantee that a message can reach a receptor timely.
Message	Any signal that an actor agent sends onto the interconnect, is encapsulated as a structured message.

Table 1. Receptor-Centric Emergence Model

Receptor-Centric Model: Receptor, Actor, Interconnect, Message

Definition 1 ***Receptor Agent***
A receptor agent is one that can receive messages from actor agents, and then conditionally update its internal states based on an intrinsic per-state control dynamics. In our model, a receptor can sense a finite set of messages; and a receptor has a finite set of states. Each state uniquely corresponds to a message type.

Definition 2 ***Actor Agent***
An actor agent is one that periodically sends messages to its environment through the interconnect. Each type of message represents a characteristic of this actor. In this paper, we define that each actor agent can only have one type characteristic; However, different agents can have the same characteristic. If a message sent by an agent cannot be received by any receptor in an environment, this actor agent is called a "dark agent".

Definition 3 ***Interconnect***
Interconnect is an asynchronous message-passing network. Any agent can send and receive messages onto the interconnect. The interconnection cannot guarantee that a message would be timely transmitted to a receptor. Any message can go wrong or be dropped before reaching its destination.

Definition 4 ***Message***
Any signal being secreted on the interconnect, from either an actor or an receptor, is encapsulated as a structured message. Each message contains a type tag.

Definition: Receptor-Centric Emergence

Definition 5 ***An RCE emergence appears iff a receptor's state is expressed explicitely.***

Receptor State Updating

Based on the RCE definition, the "Receptor State-Updating" is to formally describe how a receptor processes received messages from an actor and updates its internal states conditionally. We define *StateSet*, *MessageSet* for a receptor. and the *appreciable* field for a receptor's state is defined as: "The appreciable field is a probability value. A receptor agent processes received messages, and periodically updates this field during its life-circle. The updating criteria is determined by the overall influences that all actor agents can impose."

- Assume that a receptor receives M messages over the interconnect from a particular actor, which is defined as:
  $messageSet = \{message_1, message_2, ...message_i, ...message_M\}$.
- Assume that D is the control-dynamics for a receptor $state$, and is used for detecting a message applicable to this receptor's state or not.

Receptor State Updating Algorithm

Algorithm 1 Receptor State-Updating for One Message
Input: A receptor's state and a single message
Output: A receptor's state's appreciable value

```
function StateUpdate(state, message)
    type ← message.type.
    if type = state.type then                       ▷ sanity check in case wrong type message
        state.appreciable = D(state.appreciable, message).   ▷ process message and update the state.appreciable
    end if
    return state.appreciable
end function
```

Threshold-based Receptor Emergence Decider

Correspondingly, a decider algorithm for a receptor to process all messages from an actor is designed as: While continually updating its internal states when receiving messages from the environment, the receptor will check its intrinsic threshold value and decides whether an intrinsic emergence appears or stay recessive. In RCE model, the threshold value for a receptor defines the probability at which a start of the receptor $state.expression$ field will be turned to $TRUE$ or $FALSE$. Some important fields for a receptor's state is defined as:

- **expression**: This field is a Boolean variable. When being updated from false to true, it means that the receptor perceives an emergent characteristic.
- **threshold**: The threshold defines the probability at which the $state.expression$ will be turned to $TRUE$ or $FALSE$.

Receptor Emergence Decider Algorithm

Algorithm 2 Receptor Emergence Decider for One Actor's Influence
Input: A receptor's state and a messageSet from an actor
Output: A receptor's state's expression value

```
procedure Emergence Decider(state, messageSet)
    i = 1
    while i++ ≤ SizeOf(messageSet) do               ▷ process all messages from one actor
        message ← messageSet[i].
        state.appreciable = StateUpdate(state, message)   ▷ process a message and update state.appreciable
    end while
    if state.appreciable ≥ state.threshold then    ▷ decide whether or not turn on state.expression
        state.expression = TRUE;                     ▷ the state becomes expressed
    else
        state.expression = FALSE;                    ▷ the state stays recessive
    end if
    return state.expression.
end procedure
```

Definition: As A Whole

Definition 6 *An iterative process that a receptor handles all received messages from multiple actors in its environment, and then updates its internal states cumulatively.*

The Whole is Greater than the Sum of its Parts.

When would an emergent characteristic happen? This question has puzzled us for hundreds of years. An emergence seems to appear when the system has evolved to a critical state. The more members that make up the system, the easier it is for an emergence to occur. But, how many members are just right enough for an emergence to appear in a system? And when exactly is the critical point? So far we don't know yet.

In order to understand the above questions, we formally define the *"as a whole"* concept and semantics. And then, we prove an Emergent Characteristics Criticality (ECC) theorem, which is defined as: Given a RCE system, if the mean influence of messages by each actor agent to a receptor is $influence$, and the threshold for the receptor to flip its state's expression status is *threshold*, then consider all these actors's messages *as a whole*, an REC emergent behavior appears when the total number of actor agents reaches the $\log_{(1-influence)}(1-threshold)$.

Formal Behavior of "As a Whole"

Algorithm 3 "as a whole" process
Input: A receptor's state; the actorSet of multiple actors
Output: Process all messages from multiple actors as a whole and evaluate the state's final expression.

```
procedure As a whole(state, actorSet)
    for each actor_i ∈ actorSet do                  ▷ process multiple actors' messages
        state.expression = EmergenceDecider(state, actor_i.messageSet_i).
    end for                                          ▷ the final cumulative result is obtained
    if state.expression then                         ▷ evaluate an emergence as a whole
        Print(Emerging Characteristic Expressed: state.description);
    end if
end procedure
```

When an emergence appears? Need enough actors!

Based on the above semantics of "as a whole", we can define a receptor's *appreciable* probability distribution function as follows:

Lemma 1 *For any receptor R with a finite $StateSet$, $\forall$ $state \in StateSet$, if $\exists$ N actors that can send messages to affect $(R \to state)$, then $(R \to state).appreciable = (1 - \prod_{i=1}^{N}(1 - A_i.influence))$, where A_i is an actor agent.*

We prove by definition and induction.

1. Assume that, for an actor A_i, its influence on the receptor $(R \to state)$ is: $A_i.influence$.
2. Then the probability of no impact at all for A_i on $(R \to state)$ is: $(1 - A_i.influence)$.
3. By definition 6, processing all messages from the N actors as a whole, the overall probability of no impact at all on $(R \to state)$ is: $\prod_{i=1}^{N}(1 - A_i.influence)$.
4. Then, the overall influence of the whole N actors on the $(R \to state)$ is: $(1 - \prod_{i=1}^{N}(1 - A_i.influence))$.
5. By conjecture 1 an 3, we have: $(R \to state).appreciable = (1 - \prod_{i=1}^{N}(1 - A_i.influence))$.

With Lemma 1, we will prove the ECC theorem as below presented. The ECC is to answer the question of what the critical-point is for an emergence's expression. Before we explain the ECC theorem, we assume that for the above N actors, each of them has the same influence or have the mean value of the probability as $influence$. Therefore, the Lemma 1 is equal to $(R \to state).appreciable = (1 - (1 - influence)^N)$.

Theorem 1 *If the threshold for a receptor to flip a state's status is threshold, and the mean probability for each actor to influence this receptor's state is influence, then the ECC for this state is* $\log_{(1-influence)}(1-threshold)$.

We prove by definition and induction.

1. Assume there is a receptor R and x amount of actors in the system, and each of the x actors can independently send messages to this receptor. All the messages have the same message *type*.
2. By conjecture 3, we can use the *type* info to uniquely locate the corresponding $R \to state$.
3. By lemma 1, for this receptor R, we have:
   $state.appreciable = 1 - (1 - influence)^x$.
4. By definition 5, for an emergent characteristic appears, $state.expression$ must be set $TRUE$.
5. Then, by conjecture 2 and "Emergence Decider" algorithm, the receptor has to satisfy.
   $state.appreciable = state.threshold$.
6. Then we have:
   $1 - (1 - influence)^x = threshold$
7. we obtain $(1 - influence)^x = (1 - threshold)$.
8. Then, because $x \in$, we solve the above equation and obtain the ECC value:
   $ECC = \log_{(1-influence)}(1-threshold)$

Therefore, we prove that, even if the impact of each individual actor on a receptor is very small, a receptor's emergent characteristic still starts to appear when there have more than $\log_{(1-influence)}(1-threshold)$ actor agents in the environment.

Conclusions

The main contributions of our paper lie in the following two major aspects.

- There has been no breakthrough in the research of complex systems because it has been stuck in the discussion of philosophy and metaphysics. Most of the existing studies are fragmented, and cannot be analyzed by formal approaches. Our idea is to define the emergence from a distributed computing perspective. We define the "Receptor-Centric Emergence" as *"an emergence appears iff a receptor's state becomes expressed."*. With this RCE model and the corresponding algorithms, the "why and when" questions for the emergence are well explained.
- From Aristotle's philosophy to contemporary complex sciences, "The Whole is Greater than the Sum of its Parts" postulate has been very fascinating. And it has been argued for several decades that an emergent characteristic only happens when the system is being viewed as a whole, and the emergence only belongs to the system. However, there has been no formal definition of what "as a whole" is, except for many metaphysical discussions. In our paper, we novelly propose that the "as a whole" can be defined formally from a distributed computing perspective: *"an iterative process that a receptor handles all received messages from multiple actors in its environment, and then updates its internal states cumulatively."* With this computational semantics, we're able to much better understand an emergent behavior, both qualitatively and quantitatively.